UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/655,951	09/04/2003	George R. Cameron	3PD-M-8497 US	8894

32566 7590 01/29/2010

PATENT LAW GROUP LLP
2635 NORTH FIRST STREET
SUITE 223
SAN JOSE, CA 95134

EXAMINER	
ALAM, SHAHID AL	

ART UNIT	PAPER NUMBER
2162	

MAIL DATE	DELIVERY MODE
01/29/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte GEORGE R. CAMERON and HUEICHIAN HUANG

Appeal 2009-005447
Application 10/655,951
Technology Center 2100

Decided: January 29, 2010

Before JAY P. LUCAS, JOHN A. JEFFERY and STEPHEN C. SIU, *Administrative Patent Judges*.

SIU, *Administrative Patent Judge*.

DECISION ON APPEAL

STATEMENT OF THE CASE

This is a decision on appeal under 35 U.S.C. § 134(a) from the Examiner's rejection of claims 1-14. We have jurisdiction under 35 U.S.C. § 6(b).

We reverse.

Invention

The invention relates to data storage systems that use snapshots (Spec. 1, ¶ [0002]). The systems and methods provided create a snapshot tree structure (Spec. 5, ¶ [0026]). Snapshots are point-in-time images of data (Spec. 8, ¶ [0035]). The snapshot tree structure enables searches for data starting from a snapshot volume under consideration up to the base volume (Spec. 23-24, ¶ [0095]).

Independent claim 1 is illustrative:

> 1. A computer readable medium for a data storage device encoded with a snapshot tree structure and code for managing the snapshot tree structure to provide point-in-time backups of a base volume, wherein:
>
> the snapshot tree structure comprises:
>
> a first branch, comprising:
>
> the base volume storing a current user data;
>
> a first read-only snapshot descending from the base volume, the first read-only snapshot being created at a first time, the first read-only snapshot storing a first data of the base volume at the first time before the first data is modified in the base volume; and
>
> a second read-only snapshot descending from the first snapshot, the second read-only snapshot being created at a second time earlier than the first time,

> > the second read-only snapshot storing a second data of the base volume at the second time before the second data is modified in the base volume; and
>
> the code comprises instructions to retrieve data from the snapshot tree structure and transmitting the retrieved data to a host device.

Reference

The Examiner relies upon the following reference as evidence in support of the rejection:

Dave Hitz, et al., *File System Design for an NFS File Server Appliance*, Technical Report 3002, Rev. C 3/95, USENIX Winter 1994 (Jan. 19, 1994) ("Hitz").

Rejection

Claims 1-14 are rejected under 35 U.S.C. § 102(b) as being anticipated by Hitz.

ISSUE

The Examiner finds that "Hitz discloses that a file system (or a snapshot) can be a descend[ant] from another file system (or another snapshot)" (Ans. 8).

Appellants argue that Hitz "does not disclose that <u>any snapshot descends from another snapshot</u>" (App. Br. 5).

Issue: Did Appellants demonstrate that the Examiner erred in finding that Hitz discloses a second snapshot descending from a first snapshot?

FINDINGS OF FACT

The following Findings of Fact (FF) are shown by a preponderance of the evidence.

1. Hitz teaches that "[t]he WAFL [Write Anywhere File Layout] file system is a tree of blocks with the root inode, which describes the inode file, at the top and meta-data files and regular files underneath" (p. 8; fig 1).
2. Hitz teaches that the root inode points indirectly to inode file data blocks, which contain inodes that identify files in the file system (fig. 2). "[F]iles are made up of individual blocks and . . . large files have additional layers of indirection between the inode and the actual data blocks" (p. 9; fig. 2). "For very small files, data is stored in the inode itself in place of . . . block pointers" (p. 8). But for files that are not very small, the inode points, either directly or at varying levels of indirection, to the blocks belonging to the file (*id.*).
3. Hitz teaches that "WAFL creates a Snapshot by duplicating the root inode that describes the inode file. WAFL avoids changing blocks in a Snapshot by writing new data to new locations on disk" (p. 11; fig. 3).
4. Hitz teaches that "[t]o write a block to a new location, the pointer in the block's ancestors must be updated, which requires them to be written to new locations as well" (p. 11; fig. 4).

5. Hitz teaches that after a block update, the snapshot inode continues to point to the old inode file indirect block while the root inode points to a new inode file indirect block (fig. 4).
6. Appellants teach that "[b]ase virtual volume 2200 can be written into and read from by a user or host device 18" (Spec. 22, ¶ [0089]).
7. Appellants teach "snapshot volumes [that] may be considered to 'descend' from a base virtual volume 2200" (Spec. 22, ¶ [0090]; fig. 8). "The snapshot volumes may be grouped in branches. . . . For any given branch, the snapshot volumes extend from oldest to most recent. For example, in the branch comprising volume 2200 and snapshot volumes 2204, 2210, 2212, and 2206, snapshot volume 2206 is the oldest (created earliest in time) while snapshot 2204 is the most recent (created last in time)" (Spec. 22-23, ¶ [0091]; fig. 8).
8. Appellants teach that

   > [e]ach of the volumes in the snapshot tree from one that is under consideration up to the base volume may be analyzed in turn to see whether the data block was modified during the respective snapshots. . . . If the data block was not found in any of the snapshot volumes, then the system looks in the base virtual volume.

   (Spec. 23-24, ¶ [0095]).

PRINCIPLES OF LAW

Anticipation

In rejecting claims under 35 U.S.C. § 102, "[a] single prior art reference that discloses, either expressly or inherently, each limitation of a

claim invalidates that claim by anticipation." *Perricone v. Medicis Pharm. Corp.*, 432 F.3d 1368, 1375 (Fed. Cir. 2005) (citation omitted).

ANALYSIS

The Examiner finds that "Hitz discloses that a file system (or a snapshot) can be a descend[ant] from another file system (or another snapshot) and Hitz disclose[s] a snapshot tree structure having a first snapshot that descends from a base volume and a second snapshot that descends from the first snapshot" (Ans. 8). Appellants submit "that the tree of blocks in the WAFL file system of Hitz et al. is dissimilar from the tree structure of snapshots and base volume of claim 1" (Reply Br. 2). We agree with Appellants.

As described above, the WAFL file system of Hitz includes a tree of blocks that starts with a root inode (FF 1). The root inode indirectly points to an inode file, which comprises a set of inodes that include file data or point to the data blocks for each file (FF 2). Hitz teaches creating file system snapshots by duplicating the root inode (FF 3). To preserve the snapshot's image of the file system, WAFL writes new data to new locations on disk (FF 3), recursively writing new data to new locations for the block's ancestors (FF 4). Once this process is complete, the snapshot inode indirectly points to the old inode file data in the old locations, while the root inode indirectly points to a new inode file (FF 5).

The data within the file system, and within the file system snapshots, of Hitz is organized as a tree. However, Appellants claim a second snapshot

descending from a first snapshot (claim 1). Claim 6 (with the first and second snapshot labels reversed) and claim 11 contain similar limitations.

Appellants disclose that snapshots descend from a base volume (FF 7), where the base virtual volume can be written into and read from by a user or host device (FF 6). Snapshots can also descend from each other (FF 7). The effect of this hierarchy of snapshots is that data in each volume in a snapshot tree branch can be analyzed, starting from the one under consideration and ending at the base virtual volume, to see whether a data block was modified during the respective snapshots (FF 8).

To anticipate Appellants' invention, as described in claims 1 and 6, Hitz must teach a hierarchical relationship between snapshots. The Examiner has not demonstrated that such relationships exist among snapshots as taught by Hitz. Every snapshot in Hitz points to, either directly or indirectly, all of the data in the file system at the point at which the snapshot is created (FF 3, 4). Thus, the snapshots are independent from one another, with no snapshot descending from another snapshot. The Examiner shows that data within each snapshot is organized as a hierarchy but the Examiner does not demonstrate that Hitz teaches a hierarchy of snapshots. For at least these reasons, we find that Appellants have sustained the requisite burden on appeal in providing arguments or evidence persuasive of error in the Examiner's 35 U.S.C. § 102(b) rejection of claims 1-14.

CONCLUSIONS OF LAW

Based on the findings of facts and analysis above, we conclude that Appellants have demonstrated that the Examiner erred in finding that Hitz discloses a second snapshot descending from a first snapshot.

DECISION

We reverse the Examiner's decision rejecting claims 1-14 under 35 U.S.C. § 102(b).

REVERSED

msc

PATENT LAW GROUP LLP
2635 NORTH FIRST STREET
SUITE 223
SAN JOSE CA 95134